



Universidad  
Virtual

Desarrollo de Software

Proyecto a presentar:

Integración de Librerías en

***SIGERH***

Sistema de Gestión de Recursos Humanos

Maestría en Ingeniería y Desarrollo de Software

Mtro. Francisco Javier Rivera Magallanes

Presentado por: Elizabeth Orozco Torres

# ÍNDICE

<b>1. INTRODUCCIÓN .....</b>	<b>3</b>
<b>2. ANTECEDENTES.....</b>	<b>4</b>
<b>3. DESCRIPCIÓN GENERAL DEL PROYECTO .....</b>	<b>5</b>
3.1 OBJETIVO GENERAL .....	6
3.2 OBJETIVOS ESPECÍFICOS.....	6
<b>4. DISEÑO LÓGICO.....</b>	<b>7</b>
<b>5. DISEÑO FÍSICO.....</b>	<b>8</b>
<b>6. JUSTIFICACIÓN DEL SGBD.....</b>	<b>13</b>
<b>7. SISTEMA GESTOR DE BASE DE DATOS.....</b>	<b>14</b>
<b>8. VENTAJAS DEL SGBD FRENTE A OTROS .....</b>	<b>15</b>
<b>9. CAPTURAS DE PANTALLA DE LA BASE DE DATOS .....</b>	<b>16</b>
<b>10. INTEGRACIÓN DE LIBRERÍAS EN Sigerh.....</b>	<b>22</b>
10.1 PhpSpreadsheet.....	22
10.2 PHPMailer.....	28
10.3 TCPDF .....	31
<b>11. CONCLUSIONES .....</b>	<b>38</b>
<b>12. BIBLIOGRAFÍA.....</b>	<b>39</b>

## 1. INTRODUCCIÓN

El presente documento describe el Sistema de Gestión de Recursos Humanos (SIGERH), desarrollado para el departamento de Recursos Humanos de una empresa privada dedicada a la venta de autopartes, con dos sucursales en el Estado de Veracruz y una en la Ciudad de México. El propósito del sistema es centralizar y automatizar los procesos relacionados con el control de personal, reduciendo los costos y la carga administrativa.

El SIGERH será un sistema web que, además de realizar el cálculo de la nómina, incluirá módulos para el control de asistencias y la gestión de expedientes electrónicos del personal. Con su implementación se busca resolver una de las principales problemáticas de la empresa: el retraso en los procesos de elaboración de la nómina y el pago de impuestos.

## 2. ANTECEDENTES

La empresa enfrenta diversas problemáticas en la gestión de sus recursos humanos que afectan directamente la eficiencia operativa.

En primer lugar, el **cálculo de la nómina** se realiza actualmente mediante hojas de cálculo en Excel, lo que ha generado múltiples errores debido a fórmulas mal configuradas y a la ausencia de controles automáticos. Esta situación provoca inconsistencias en los pagos y retrasa los procesos contables.

En segundo lugar, el **registro de asistencia** se lleva de manera manual en una libreta donde cada empleado anota su hora de entrada y salida. Este método resulta poco confiable y dificulta la verificación del cumplimiento horario, generando discrepancias en los reportes de asistencia.

Finalmente, los **expedientes del personal** se conservan únicamente en formato físico. La documentación presenta desorganización, falta de información y riesgo de extravío, lo que deriva en errores administrativos, multas por declaraciones incorrectas y una gestión ineficiente del capital humano.

Estas deficiencias justifican la necesidad de implementar un sistema integral que automatice y centralice los procesos de nómina, asistencia y expedientes laborales, garantizando un control más preciso y una operación administrativa eficiente.

### 3. DESCRIPCIÓN GENERAL DEL PROYECTO

El Sistema de Gestión de Recursos Humanos (SIGERH) será una aplicación web desarrollada con **HTML, PHP y JavaScript**, utilizando **MySQL** como sistema gestor de base de datos. Su finalidad es proporcionar al personal del departamento de Recursos Humanos una herramienta que centralice el control de los trabajadores desde su ingreso a la empresa, facilitando la gestión de expedientes electrónicos, el cálculo de la nómina y la consulta de registros laborales.

El sistema integrará las siguientes funcionalidades principales:

1. **Captura de datos generales** del empleado y anexo de documentos para conformar su expediente electrónico.
2. **Registro de nuevos usuarios del sistema**, asignando roles y permisos correspondientes.
3. **Cálculo automatizado de la nómina**, con integración de incidencias laborales.
4. **Consulta y actualización de expedientes** del personal.
5. **Generación de reportes** sobre asistencia, incidencias (retardos y faltas), nómina y pagos de impuestos.

El SIGERH contará con tres tipos de usuarios o roles:

- **Trabajador:** podrá consultar sus registros de entrada y salida, así como la información relacionada con su nómina.
- **Capturista:** integrante del área de Recursos Humanos encargado de registrar datos personales, horarios, sueldos, documentos digitalizados y altas de usuarios.
- **Supervisor:** jefe del departamento de Recursos Humanos responsable de la administración de permisos, supervisión de actividades y generación de reportes globales.

Con esta estructura, el sistema busca optimizar los procesos administrativos y mejorar la precisión en la gestión del personal, reduciendo los errores derivados del manejo manual de información.

### **3.1 OBJETIVO GENERAL**

Desarrollar un sistema integral de gestión de recursos humanos que automatice los procesos de cálculo de nómina, control de asistencia y administración de expedientes electrónicos del personal, con el propósito de optimizar la eficiencia operativa y reducir los errores administrativos en la empresa.

### **3.2 OBJETIVOS ESPECÍFICOS**

1. Automatizar el proceso de cálculo y emisión de la nómina, garantizando precisión en los pagos y cumplimiento en los plazos establecidos.
2. Implementar un expediente electrónico para cada trabajador, que concentre y resguarde de forma segura su información personal, laboral y documental.
3. Optimizar el control de asistencia del personal mediante el registro digital de entradas, salidas e incidencias, fortaleciendo la confiabilidad de los reportes.

## 4. DISEÑO LÓGICO



## 5. DISEÑO FÍSICO

### Estructura de tabla para la tabla `empleados`

```
CREATE TABLE `empleados` (  
  `id` int(11) NOT NULL,  
  `codigo_empleado` varchar(20) DEFAULT NULL,  
  `nombre` varchar(100) NOT NULL,  
  `apellido` varchar(100) NOT NULL,  
  `documento` varchar(20) DEFAULT NULL,  
  `correo` varchar(100) DEFAULT NULL,  
  `telefono` varchar(20) DEFAULT NULL,  
  `direccion` varchar(255) DEFAULT NULL,  
  `fecha_nacimiento` date DEFAULT NULL,  
  `fecha_ingreso` date NOT NULL,  
  `sucursal_id` int(11) DEFAULT NULL,  
  `puesto_id` int(11) DEFAULT NULL,  
  `puesto` varchar(100) DEFAULT NULL,  
  `salario` decimal(10,2) DEFAULT 0.00,  
  `estado_empleado` enum('activo','inactivo','suspendido') DEFAULT 'activo',  
  `fecha_creacion` timestamp NOT NULL DEFAULT current_timestamp(),  
  `usuario_creacion` varchar(50) DEFAULT NULL,  
  `usuario_modificacion` varchar(50) DEFAULT NULL,  
  `fecha_modificacion` timestamp NULL DEFAULT NULL,  
  `activo` tinyint(1) DEFAULT 1  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

### Estructura de tabla para la tabla `expedientes`

```
CREATE TABLE `expedientes` (  
  `id` int(11) NOT NULL,  
  `empleado_id` int(11) NOT NULL,  
  `curriculum` varchar(255) DEFAULT NULL,  
  `solicitud_empleo` varchar(255) DEFAULT NULL,  
  `fotografia` varchar(255) DEFAULT NULL,  
  `constancia_estudios` varchar(255) DEFAULT NULL,  
  `curp` varchar(255) DEFAULT NULL,  
  `comprobante_domicilio` varchar(255) DEFAULT NULL,  
  `rfc` varchar(255) DEFAULT NULL,  
  `numero_seguridad_social` varchar(255) DEFAULT NULL,  
  `constancia_medica` varchar(255) DEFAULT NULL,  
  `otros_documentos` text DEFAULT NULL,  
  `fecha_creacion` timestamp NOT NULL DEFAULT current_timestamp(),  
  `activo` tinyint(1) DEFAULT 1  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```



### Estructura de tabla para la tabla `nominas`

```
CREATE TABLE `nominas` (  
  `id` int(11) NOT NULL,  
  `empleado_id` int(11) NOT NULL,  
  `periodo_inicio` date NOT NULL,  
  `periodo_fin` date NOT NULL,  
  `fecha_pago` date NOT NULL,  
  `salario_base` decimal(10,2) NOT NULL,  
  `bonificaciones` decimal(10,2) DEFAULT 0.00,  
  `deducciones` decimal(10,2) DEFAULT 0.00,  
  `salario_neto` decimal(10,2) NOT NULL,  
  `metodo_pago` varchar(50) DEFAULT NULL,  
  `comentarios` text DEFAULT NULL,  
  `fecha_creacion` timestamp NOT NULL DEFAULT current_timestamp(),  
  `usuario_creacion` varchar(50) DEFAULT NULL,  
  `usuario_modificacion` varchar(50) DEFAULT NULL,  
  `fecha_modificacion` timestamp NULL DEFAULT NULL,  
  `activo` tinyint(1) DEFAULT 1  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

### Estructura de tabla para la tabla `nomina\_historial`

```
CREATE TABLE `nomina_historial` (  
  `id` int(11) NOT NULL,  
  `nomina_id` int(11) NOT NULL,  
  `empleado_id` int(11) NOT NULL,  
  `fecha_cambio` timestamp NOT NULL DEFAULT current_timestamp(),  
  `salario_base` decimal(10,2) DEFAULT NULL,  
  `bonificaciones` decimal(10,2) DEFAULT NULL,  
  `deducciones` decimal(10,2) DEFAULT NULL,  
  `salario_neto` decimal(10,2) DEFAULT NULL,  
  `motivo_cambio` varchar(255) DEFAULT NULL,  
  `usuario_responsable` varchar(50) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

### Estructura de tabla para la tabla `sucursales`

```
CREATE TABLE `sucursales` (  
  `id` int(11) NOT NULL,  
  `nombre` varchar(100) NOT NULL,  
  `direccion` varchar(255) NOT NULL,  
  `telefono` varchar(20) DEFAULT NULL,  
  `correo` varchar(100) DEFAULT NULL,  
  `encargado` varchar(100) DEFAULT NULL,  
  `estado` varchar(100) DEFAULT NULL,  
  `ciudad` varchar(100) DEFAULT NULL,
```

```
`codigo_postal` varchar(10) DEFAULT NULL,  
`fecha_creacion` timestamp NOT NULL DEFAULT current_timestamp(),  
`activo` tinyint(1) DEFAULT 1  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

#### Estructura de tabla para la tabla `usuarios`

```
CREATE TABLE `usuarios` (  
  `id` int(11) NOT NULL,  
  `nombre_usuario` varchar(50) NOT NULL,  
  `password` varchar(255) NOT NULL,  
  `nombre_completo` varchar(100) NOT NULL,  
  `rol` enum('admin','rrhh','contabilidad','invitado') DEFAULT 'invitado',  
  `correo` varchar(100) DEFAULT NULL,  
  `fecha_creacion` timestamp NOT NULL DEFAULT current_timestamp(),  
  `usuario_creacion` varchar(50) DEFAULT NULL,  
  `usuario_modificacion` varchar(50) DEFAULT NULL,  
  `fecha_modificacion` timestamp NULL DEFAULT NULL,  
  `activo` tinyint(1) DEFAULT 1  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

#### Estructura de tabla para la tabla `roles`

```
CREATE TABLE `roles` (  
  `id` int(11) NOT NULL,  
  `nombre_rol` varchar(50) NOT NULL,  
  `descripcion` varchar(255) DEFAULT NULL,  
  `fecha_creacion` timestamp NOT NULL DEFAULT current_timestamp(),  
  `activo` tinyint(1) DEFAULT 1  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

#### Estructura de tabla para la tabla `puestos`

```
CREATE TABLE `puestos` (  
  `id` int(11) NOT NULL,  
  `nombre_puesto` varchar(100) NOT NULL,  
  `descripcion` varchar(255) DEFAULT NULL,  
  `salario_base` decimal(10,2) DEFAULT 0.00,  
  `nivel` varchar(50) DEFAULT NULL,  
  `departamento` varchar(100) DEFAULT NULL,  
  `fecha_creacion` timestamp NOT NULL DEFAULT current_timestamp(),  
  `activo` tinyint(1) DEFAULT 1  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

#### Estructura de tabla para la tabla `permisos`

```
CREATE TABLE `permisos` (  
  `id` int(11) NOT NULL,  
  `empleado_id` int(11) NOT NULL,  
  `fecha_inicio` datetime NOT NULL,  
  `fecha_fin` datetime NOT NULL,  
  `tipo` enum('2h','cita_medica','otros') NOT NULL,  
  `comentarios` text DEFAULT NULL,  
  `estado` enum('pendiente','aprobado','rechazado') DEFAULT 'pendiente',  
  `fecha_creacion` timestamp NOT NULL DEFAULT current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

#### Estructura de tabla para la tabla `justificaciones`

```
CREATE TABLE `justificaciones` (  
  `id` int(11) NOT NULL,  
  `empleado_id` int(11) NOT NULL,  
  `fecha_falta` date NOT NULL,  
  `tipo` enum('falta','retardo') NOT NULL,  
  `documento` varchar(255) DEFAULT NULL,  
  `comentarios` text DEFAULT NULL,  
  `fecha_creacion` timestamp NOT NULL DEFAULT current_timestamp(),  
  `estado` enum('pendiente','aprobado','rechazado') DEFAULT 'pendiente'  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

#### Estructura de tabla para la tabla `vacaciones`

```
CREATE TABLE `vacaciones` (  
  `id` int(11) NOT NULL,  
  `empleado_id` int(11) NOT NULL,  
  `fecha_inicio` date NOT NULL,  
  `fecha_fin` date NOT NULL,  
  `dias_disfrutados` int(11) NOT NULL,  
  `estado` enum('pendiente','aprobado','rechazado') DEFAULT 'pendiente',  
  `comentarios` text DEFAULT NULL,  
  `fecha_creacion` timestamp NOT NULL DEFAULT current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

#### Estructura de tabla para la tabla `bonificaciones\_nomina`

```
CREATE TABLE `bonificaciones_nomina` (  
  `id` int(11) NOT NULL,  
  `nomina_id` int(11) NOT NULL,  
  `tipo` varchar(100) NOT NULL,  
  `monto` decimal(10,2) NOT NULL,  
  `descripcion` text DEFAULT NULL,  
  `fecha_creacion` timestamp NOT NULL DEFAULT current_timestamp(),
```

```
`activo` tinyint(1) DEFAULT 1  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

#### Estructura de tabla para la tabla `che cadas`

```
CREATE TABLE `che cadas` (  
  `id` int(11) NOT NULL,  
  `empleado_id` int(11) NOT NULL,  
  `fecha` date NOT NULL,  
  `hora` time NOT NULL,  
  `tipo` enum('entrada','salida','entrada_comida','salida_definitiva') NOT NULL,  
  `sucursal_id` int(11) DEFAULT NULL,  
  `comentarios` text DEFAULT NULL,  
  `fecha_creacion` timestamp NOT NULL DEFAULT current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

#### Estructura de tabla para la tabla `deducciones\_nomina`

```
CREATE TABLE `deducciones_nomina` (  
  `id` int(11) NOT NULL,  
  `nomina_id` int(11) NOT NULL,  
  `tipo` varchar(100) NOT NULL,  
  `monto` decimal(10,2) NOT NULL,  
  `descripcion` text DEFAULT NULL,  
  `fecha_creacion` timestamp NOT NULL DEFAULT current_timestamp(),  
  `activo` tinyint(1) DEFAULT 1  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

## 6. JUSTIFICACIÓN DEL SGBD




El modelo elegido para la base de datos del sistema SIGERH es el modelo relacional, implementado en MySQL, por ser el más adecuado para sistemas administrativos con requerimientos de integridad, consistencia y escalabilidad. Este modelo permite representar de manera clara las relaciones entre los distintos componentes del sistema —trabajadores, nómina, asistencias y expedientes— mediante el uso de claves primarias y foráneas, garantizando la integridad referencial.

Dado que el SIGERH debe manejar datos sensibles (sueldos, deducciones, impuestos, expedientes digitales), el modelo relacional ofrece control transaccional (ACID), indispensable para evitar errores o pérdidas de información durante los cálculos de nómina. Además, la estructura tabular facilita la ejecución de consultas SQL complejas y la generación de reportes dinámicos para los módulos administrativos.

Otro argumento clave es que MySQL, como motor relacional, ofrece seguridad, rendimiento y compatibilidad multiplataforma, lo que permite integrar fácilmente el backend desarrollado en PHP y las consultas desde JavaScript (AJAX) sin comprometer el desempeño. Su arquitectura modular permite escalar de manera progresiva —por ejemplo, agregar módulos de prestaciones o auditoría— sin modificar la estructura general. En resumen, el modelo relacional basado en MySQL resulta el más idóneo por su fiabilidad, flexibilidad, eficiencia en el manejo de relaciones y por ser una solución probada y estable para sistemas empresariales web como el SIGERH.

## 7. SISTEMA GESTOR DE BASE DE DATOS

El sistema puede ejecutarse y visualizarse en:

-  Sistema Gestor de Base de Datos: MySQL Server (versión 8.0 o superior)
-  Entorno de desarrollo:
  - phpMyAdmin (para gestión visual desde navegador)
  - MySQL Workbench (para modelado EER, ejecución de queries y administración avanzada)
  - XAMPP / LAMPP / WAMP como servidores locales (para integrar Apache + PHP + MySQL).
-  Lenguaje de acceso: PHP (a través de mysqli o PDO) para la conexión al sistema web.

El Sistema Gestor de Base de Datos (SGBD) MySQL fue elegido para el proyecto SIGERH por razones técnicas, operativas y de eficiencia, que lo convierten en la opción más adecuada para el tipo de sistema propuesto, MySQL es el modelo ideal para la implementación de la base de datos del proyecto SIGERH debido a su rendimiento, compatibilidad, seguridad, escalabilidad y bajo costo de implementación. Además, su naturaleza relacional permite representar de manera eficiente las entidades del sistema —empleados, nómina, asistencia y expedientes— y mantener relaciones consistentes entre ellas mediante el uso de llaves primarias y foráneas, garantizando la integridad referencial de los datos a lo largo del ciclo de vida del sistema.

## 8. VENTAJAS DEL SGBD FRENTE A OTROS

MySQL fue elegido como Sistema Gestor de Base de Datos (SGBD) para el proyecto SIGERH por ser una solución robusta, ampliamente utilizada y con soporte multiplataforma. Se adapta perfectamente a aplicaciones web desarrolladas en **PHP, HTML y JavaScript**, ofreciendo un equilibrio ideal entre rendimiento, seguridad y facilidad de uso.

A continuación, se presenta una tabla comparativa con otros SGBD, que fortalece la elección realizada:

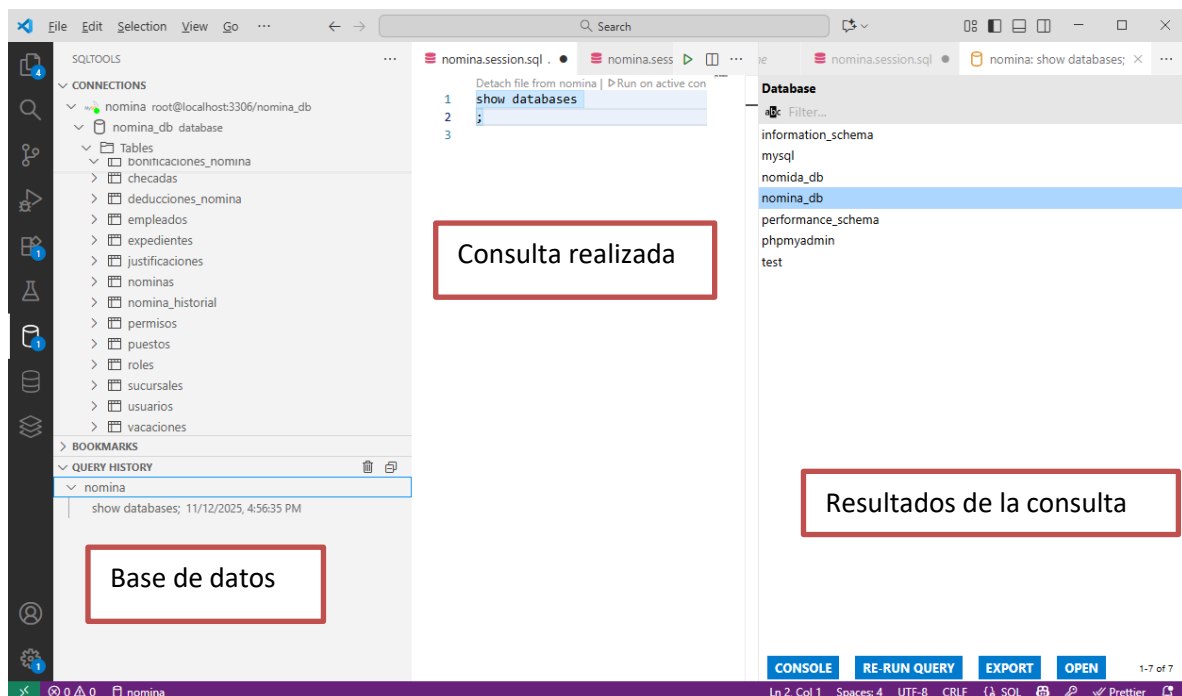
Característica	MySQL	SQL Server	Oracle	PostgreSQL
<b>Costo</b>	Gratuito (GPL)	Licencia comercial	Licencia costosa	Gratuito
<b>Compatibilidad web</b>	Alta con PHP / Apache	Limitada a entornos Microsoft	Compleja	Alta
<b>Rendimiento</b>	Excelente en consultas de lectura	Alto, pero requiere más recursos	Muy alto, pero complejo	Muy alto
<b>Escalabilidad</b>	Alta (clustering y replicación)	Alta	Muy alta	Muy alta
<b>Facilidad de administración</b>	Muy buena (phpMyAdmin, Workbench)	Requiere SQL Management Studio	Requiere DBA	Requiere experiencia técnica
<b>Comunidad</b>	Amplia y activa	Limitada	Limitada	Activa

En conclusión, MySQL ofrece todas las capacidades necesarias para SIGERH —transacciones ACID, integridad referencial, vistas, procedimientos almacenados y replicación— sin incurrir en costos adicionales. Además, su integración con **PHP** y su interfaz gráfica en **phpMyAdmin** simplifican tanto el desarrollo como la administración del sistema.

## 9. CAPTURAS DE PANTALLA DE LA BASE DE DATOS

Las capturas de pantalla que se muestran a continuación fueron tomadas en Visual Studio Code. La pantalla se dividió en tres secciones con el propósito de mostrar, de manera simultánea, las columnas de la tabla, la consulta realizada sobre una tabla y los datos que esta contiene. La base de datos denominada “nomina\_db” cuenta con un total de 14 tablas, de las cuales únicamente 9 poseen registros; de esas tablas se realizaron las capturas de pantalla correspondientes.

Vista del VS Code de la base de dato:





## Vista de la Tabla empleados:

The screenshot shows a database management tool interface. On the left, the 'DATABASE' pane shows the 'nomina' database with various tables. The 'empleados' table is selected, and its columns are listed: id, codigo\_empleado, nombre, apellido, documento, correo, telefono, fecha\_nacimiento, fecha\_ingreso, sucursal\_id, puesto, salario, estado\_empleado, fecha\_creacion, and usuario\_creacion. The main pane displays the 'empleados' table data, showing 15 rows of employee information.

id	codigo_empleado	nombre	apellido	documento	correo	telefono
1	EMP001	Juan	Hernández	12345678	juan.hernandez@empresa.c	555-111-2
2	EMP002	Maria	López	98765432	maria.lopez@empresa.com	555-333-4
3	EMP003	Carlos	Pérez	11223344	carlos.perez@empresa.com	555-777-8
4	EMP004	Lucía	Martínez	99887766	lucia.martinez@empresa.co	555-100-0
5	EMP005	Pedro	Gómez	88776655	pedro.gomez@empres@r	555-100-0
6	EMP006	Ana	Torres	77889911	ana.torres@empresa.com	555-100-0
7	EMP007	Jorge	Ramírez	55443322	jorge.ramirez@empresa.cor	555-100-0
8	EMP008	Patricia	Cruz	66554433	patricia.cruz@empresa.com	555-100-0
9	EMP009	Fernando	Santos	22334455	fernando.santos@empresa	555-100-0
10	EMP010	Claudia	Ríos	99112233	claudia.rios@empresa.com	555-100-0
11	EMP011	Luis	Morales	33445566	luis.morales@empresa.com	555-100-0
12	EMP012	Carmen	Flores	44332211	carmen.flores@empresa.co	555-100-0
13	EMP013	Andrés	Pérez	55667788	andres.perez@empresa.com	555-100-0
14	EMP014	Rocío	Navarro	66778899	rocio.navarro@empresa.cor	555-100-0
15	EMP015	Sergio	Vega	77889900	sergio.vega@empresa.com	555-100-0

## Vista de la Tabla sucursales:

The screenshot shows the same database management tool interface, but with the 'sucursales' table selected. The columns listed are: id, nombre, direccion, telefono, correo, encargado, estado, ciudad, codigo\_postal, fecha\_creacion, and activo. The main pane displays the 'sucursales' table data, showing 3 rows of branch information.

id	nombre	direccion	telefono	correo	encargado	estado
1	Sucursal Central	Av. Principal #123, Col. Cent	555-123-4567	central@empresa.com	Laura Méndez	Estado Capital
2	Sucursal Norte	Calle 45 #89, Col. Industrial	555-987-6543	norte@empresa.com	Carlos Pérez	Estado Norte
3	Sucursal Sur	Av. Libertad #250	555-321-7890	sur@empresa.com	Ana Torres	Estado Sur

## Vista de la Tabla nominas:

The screenshot shows a database management tool interface. On the left, the 'nominas' table is selected, showing its columns: id, empleado\_id, periodo\_inicio, periodo\_fin, fecha\_pago, salario\_base, bonificaciones, deducciones, salario\_neto, metodo\_pago, comentarios, fecha\_creacion, usuario\_creacion, usuario\_modificacion, fecha\_modificacion, and activo. The main pane displays the table data with columns: id, empleado, periodo\_inicio, periodo\_fin, fecha\_pago, salario\_base, and bonificaciones. The data is limited to 20 rows.

id	empleado	periodo_inicio	periodo_fin	fecha_pago	salario_base	bonificaciones
1	1	2025-10-01	2025-10-15	2025-10-16	15000.00	1000.00
2	2	2025-10-01	2025-10-15	2025-10-16	18000.00	1200.00
3	3	2025-10-01	2025-10-15	2025-10-16	20000.00	2000.00
4	4	2025-10-01	2025-10-15	2025-10-16	15000.00	800.00
5	5	2025-10-01	2025-10-15	2025-10-16	18000.00	500.00
6	6	2025-10-01	2025-10-15	2025-10-16	12000.00	600.00
7	7	2025-10-01	2025-10-15	2025-10-16	20000.00	1500.00
8	8	2025-10-01	2025-10-15	2025-10-16	15000.00	400.00
9	9	2025-10-01	2025-10-15	2025-10-16	12000.00	700.00
10	10	2025-10-01	2025-10-15	2025-10-16	20000.00	900.00
11	11	2025-10-01	2025-10-15	2025-10-16	18000.00	1000.00
12	12	2025-10-01	2025-10-15	2025-10-16	15000.00	500.00
13	13	2025-10-01	2025-10-15	2025-10-16	12000.00	800.00
14	14	2025-10-01	2025-10-15	2025-10-16	18000.00	700.00
15	15	2025-10-01	2025-10-15	2025-10-16	15000.00	600.00

## Vista de la Tabla puestos:

The screenshot shows a database management tool interface. On the left, the 'puestos' table is selected, showing its columns: id, nombre\_puesto, descripcion, salario\_base, nivel, departamento, fecha\_creacion, and activo. The main pane displays the table data with columns: id, nombre\_puesto, descripcion, salario\_base, nivel, departamento, and fecha\_creacion. The data is limited to 5 rows.

id	nombre_puesto	descripcion	salario_base	nivel	departamento	fecha_creacion
1	Gerente General	Responsable de la dirección	30000.00	Dirección	Administración	2025-11-11 1
2	Contador General	Encargado de la contabilidad	20000.00	Profesional	Contabilidad	2025-11-11 1
3	Analista de Nómina	Gestiona el cálculo y pago c	15000.00	Operativo	Recursos Humanos	2025-11-11 1
4	Auxiliar Administrativo	Apoya las tareas administra	12000.00	Operativo	Administración	2025-11-11 1
5	Encargado de Sucursal	Coordina operaciones en ur	18000.00	Supervisión	Operaciones	2025-11-11 1

## Vista de la Tabla roles:

The screenshot shows the SQL Studio interface. On the left, the 'DATABASE' pane shows the 'nomina\_db' database with a tree view of tables. The 'roles' table is selected, showing its columns: id (int(11)), nombre\_rol (varchar(50)), descripcion (varchar(255)), fecha\_creacion (timestamp), and activo (tinyint(1)). The main pane shows the 'Search Results' for the query 'SELECT \* FROM roles LIMIT 100'. The results are as follows:

id	nombre_rol	descripcion	fecha_creacion	activo
1	admin	Acceso total al sistema	2025-11-11 11:24:48	1
2	rrhh	Gestiona empleados y nómi	2025-11-11 11:24:48	1
3	contabilidad	Revisa pagos y reportes fin	2025-11-11 11:24:48	1
4	invitado	Acceso limitado solo de lect	2025-11-11 11:24:48	1

## Vista de la Tabla bonificaciones\_nomina:

The screenshot shows the SQL Studio interface. On the left, the 'DATABASE' pane shows the 'nomina\_db' database with a tree view of tables. The 'bonificaciones\_nomina' table is selected, showing its columns: id (int(11)), nomina\_id (int(11)), tipo (varchar(100)), monto (decimal(10,2)), descripcion (text), fecha\_creacion (timestamp), and activo (tinyint(1)). The main pane shows the 'Search Results' for the query 'SELECT \* FROM bonificaciones\_nomina LIMIT 100'. The results are as follows:

id	nomina_id	tipo	monto	descripcion	fecha_creacion
1	1	Bono productividad	1000.00	Bono por productividad alti	2025-11-11 13:17:42
2	2	Bono puntualidad	1200.00	Bono por puntualidad	2025-11-11 13:17:42
3	3	Bono desempeño	2000.00	Bono por desempeño exce	2025-11-11 13:17:42
4	4	Bono asistencia	800.00	Bono por asistencia perfect	2025-11-11 13:17:42
5	5	Bono liderazgo	500.00	Bono por liderazgo	2025-11-11 13:17:42
6	6	Bono productividad	600.00	Bono por productividad alti	2025-11-11 13:17:42
7	7	Bono desempeño	1500.00	Bono por desempeño	2025-11-11 13:17:42
8	8	Bono asistencia	400.00	Bono asistencia	2025-11-11 13:17:42
9	9	Bono puntualidad	700.00	Bono puntualidad	2025-11-11 13:17:42
10	10	Bono productividad	900.00	Bono productividad	2025-11-11 13:17:42
11	11	Bono desempeño	1000.00	Bono desempeño	2025-11-11 13:17:42
12	12	Bono asistencia	500.00	Bono asistencia	2025-11-11 13:17:42
13	13	Bono puntualidad	800.00	Bono puntualidad	2025-11-11 13:17:42
14	14	Bono productividad	700.00	Bono productividad	2025-11-11 13:17:42
15	15	Bono desempeño	600.00	Bono desempeño	2025-11-11 13:17:42

## Vista de la Tabla checadas:

The screenshot shows a database management tool interface. On the left, the 'DATABASE' pane shows the 'nomina\_db' database with a table 'checadas' containing 87 rows. The 'Columns' pane for 'checadas' lists: id int(11), empleado\_id int(11), fecha date, hora time, tipo enum('entrada','salida','entrada\_comida','salida\_definitiva'), sucursal\_id int(11), comentarios text, and fecha\_creacion timestamp. The main pane shows the 'checadas' table with a query: `SELECT * FROM checadas LIMIT 100`. The results show 40 rows of data.

	id	empleado_id	fecha	hora	tipo	sucursal_id	comentarios
>	26	3	2025-11-03	13:00:00	salida	(NULL)	(NULL)
>	27	3	2025-11-03	14:00:00	entrada_comida	(NULL)	(NULL)
>	28	3	2025-11-03	18:00:00	salida_definitiva	(NULL)	(NULL)
>	29	4	2025-11-03	08:15:00	entrada	(NULL)	(NULL)
>	30	4	2025-11-03	13:00:00	salida	(NULL)	(NULL)
>	31	4	2025-11-03	14:00:00	entrada_comida	(NULL)	(NULL)
>	32	4	2025-11-03	18:10:00	salida_definitiva	(NULL)	(NULL)
>	33	5	2025-11-03	08:00:00	entrada	(NULL)	(NULL)
>	34	5	2025-11-03	13:00:00	salida	(NULL)	(NULL)
>	35	5	2025-11-03	14:00:00	entrada_comida	(NULL)	(NULL)
>	36	5	2025-11-03	18:00:00	salida_definitiva	(NULL)	(NULL)
>	37	6	2025-11-03	08:05:00	entrada	(NULL)	(NULL)
>	38	6	2025-11-03	13:00:00	salida	(NULL)	(NULL)
>	39	6	2025-11-03	14:00:00	entrada_comida	(NULL)	(NULL)
>	40	6	2025-11-03	18:05:00	salida_definitiva	(NULL)	(NULL)

## Vista de la Tabla deducciones\_nomina:

The screenshot shows the same database management tool interface, but now displaying the 'deducciones\_nomina' table. The 'Columns' pane for 'deducciones\_nomina' lists: id int(11), nomina\_id int(11), tipo varchar(100), monto decimal(10,2), descripcion text, fecha\_creacion timestamp, and activo tinyint(1). The main pane shows the 'deducciones\_nomina' table with a query: `SELECT * FROM deducciones_nomina LIMIT 100`. The results show 15 rows of data.

	id	nomina_id	tipo	monto	descripcion	fecha_creacion	activo
>	1	1	ISR	1200.00	Impuesto sobre la renta	2025-11-11 13:20:41	1
>	2	2	IMSS	500.00	Seguridad social	2025-11-11 13:20:41	1
>	3	3	INFONAVIT	800.00	Crédito INFONAVIT	2025-11-11 13:20:41	1
>	4	4	ISR	1000.00	Impuesto sobre la renta	2025-11-11 13:20:41	1
>	5	5	IMSS	600.00	Seguridad social	2025-11-11 13:20:41	1
>	6	6	INFONAVIT	700.00	Crédito INFONAVIT	2025-11-11 13:20:41	1
>	7	7	ISR	1100.00	Impuesto sobre la renta	2025-11-11 13:20:41	1
>	8	8	IMSS	400.00	Seguridad social	2025-11-11 13:20:41	1
>	9	9	INFONAVIT	900.00	Crédito INFONAVIT	2025-11-11 13:20:41	1
>	10	10	ISR	1300.00	Impuesto sobre la renta	2025-11-11 13:20:41	1
>	11	11	IMSS	550.00	Seguridad social	2025-11-11 13:20:41	1
>	12	12	INFONAVIT	650.00	Crédito INFONAVIT	2025-11-11 13:20:41	1
>	13	13	ISR	1200.00	Impuesto sobre la renta	2025-11-11 13:20:41	1
>	14	14	IMSS	700.00	Seguridad social	2025-11-11 13:20:41	1
>	15	15	INFONAVIT	800.00	Crédito INFONAVIT	2025-11-11 13:20:41	1

## Vista de la Tabla usuarios:

The screenshot shows a database management interface with the following components:

- Left Panel (Database Explorer):** Displays the database structure. The 'usuarios' table is selected, showing its columns: id (int(11)), nombre\_usuario (varchar(50)), password (varchar(255)), nombre\_completo (varchar(100)), rol (enum('admin','rrhh','contabilidad',...)), correo (varchar(100)), fecha\_creacion (timestamp), usuario\_creacion (varchar(50)), usuario\_modificacion (varchar(50)), fecha\_modificacion (timestamp), and activo (tinyint(1)).
- SQL Editor:** Contains the query: 

```
SELECT * FROM usuarios LIMIT 100;
```
- Results Panel:** Displays the query results in a table format. The table has 6 columns: id, nombre\_usuario, password, nombre\_completo, rol, and a truncated 'activo' column. The results show 21 rows of user data.

id	nombre_usuario	password	nombre_completo	rol	activo
1	admin	\$2y\$10\$7qM9Y9YQzZzVnh	Administrador del Sistema	admin	adm
7	usuario1	\$2y\$10\$Qpw1	Lucía Martínez	rrhh	luci
8	usuario2	\$2y\$10\$Qpw2	Pedro Gómez	contabilidad	ped
9	usuario3	\$2y\$10\$Qpw3	Ana Torres	rrhh	ana
10	usuario4	\$2y\$10\$Qpw4	Jorge Ramírez	invitado	jorg
11	usuario5	\$2y\$10\$Qpw5	Patricia Cruz	rrhh	patr
12	usuario6	\$2y\$10\$Qpw6	Fernando Santos	contabilidad	fern
13	usuario7	\$2y\$10\$Qpw7	Claudia Ríos	rrhh	clau
14	usuario8	\$2y\$10\$Qpw8	Luis Morales	contabilidad	luis
15	usuario9	\$2y\$10\$Qpw9	Carmen Flores	rrhh	carm
16	usuario10	\$2y\$10\$Qpw10	Andrés Pérez	invitado	and
17	usuario11	\$2y\$10\$Qpw11	Rocío Navarro	rrhh	roci
18	usuario12	\$2y\$10\$Qpw12	Sergio Vega	contabilidad	ser
19	usuario13	\$2y\$10\$Qpw13	Adriana Reyes	rrhh	adri
20	usuario14	\$2y\$10\$Qpw14	David Mendoza	invitado	dav
21	usuario15	\$2y\$10\$Qpw15	Gabriela Suárez	rrhh	gab

## 10. INTEGRACIÓN DE LIBRERÍAS EN SIGERH

Para la implementación del Sistema de Gestión de Recursos Humanos (SIGERH), se integraron diversas librerías que permiten mejorar la funcionalidad del sistema, optimizar procesos administrativos y facilitar la generación de reportes requeridos por el departamento de Recursos Humanos. A continuación, se describen 3 librerías que considere son las más funcionales para el proyecto, justificando su incorporación y mostrando su implementación práctica dentro del proyecto.

### 10.1 PhpSpreadsheet (Exportación de Reportes en Excel)

#### Descripción

PhpSpreadsheet es una librería PHP que permite crear, leer y modificar archivos de Excel (.xlsx, .xls) y otros formatos como CSV. Es la librería más utilizada en sistemas administrativos para generar reportes profesionalmente desde la base de datos.

#### Justificación

El sistema SIGERH, requiere generar reportes editables de los empleados, nóminas, asistencias, deducciones, entre otros. Exportar estos reportes a Excel es fundamental para:

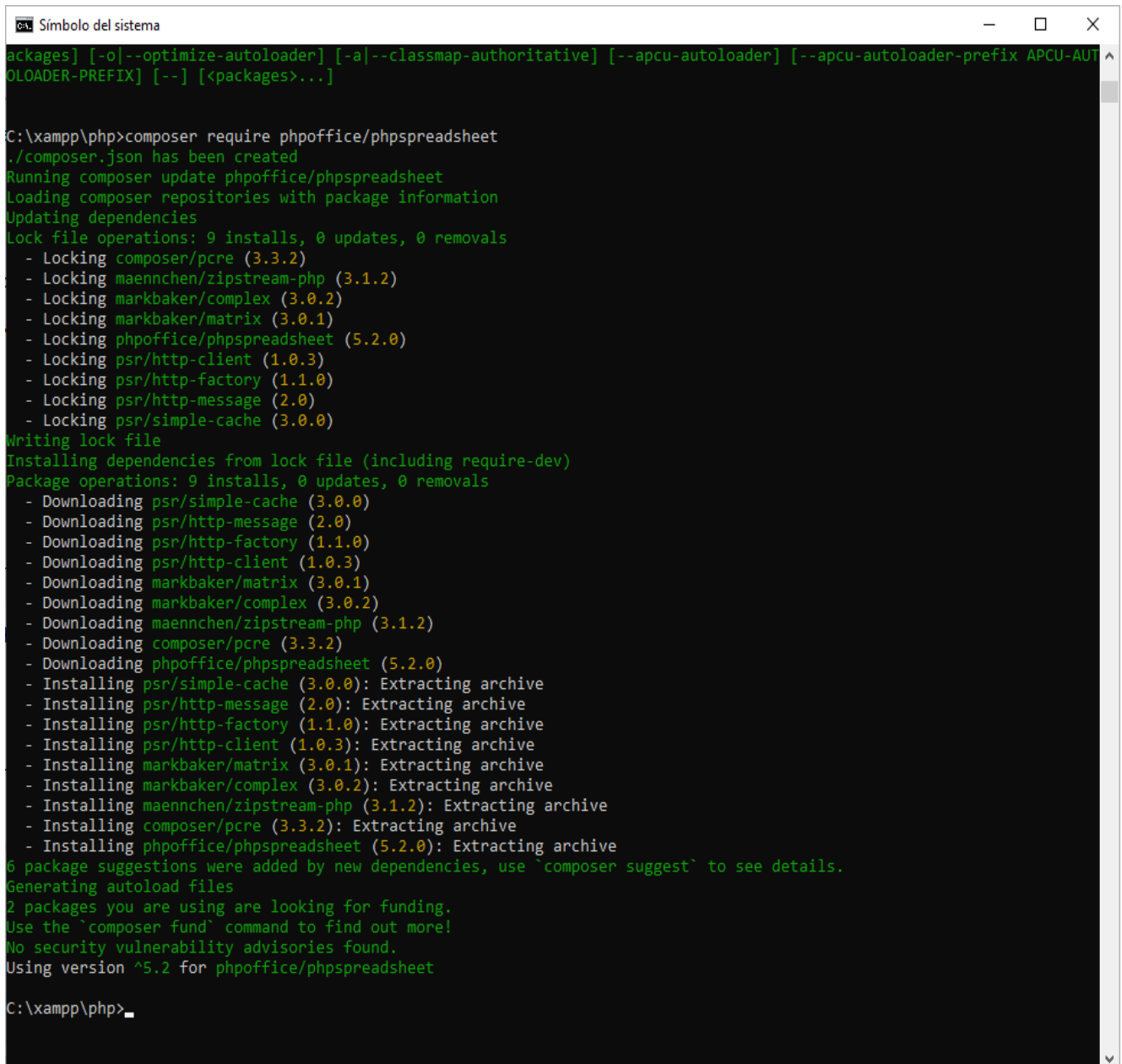
- ✚ entregar información al área contable,
- ✚ realizar auditorías internas,
- ✚ reportar incidencias y pagos,
- ✚ respaldar información semanal o quincenal.

PhpSpreadsheet permite crear reportes dinámicos con buena presentación, encabezados, estilos de celda y filtros.

## Instalación de la librería:

PhpSpreadsheet se instaló mediante el uso de composer, en la siguiente pantalla se muestra el proceso de instalación mediante el siguiente comando:

**composer require phpooffice/phpspreadsheet**



```
Símbolo del sistema
[...-o] [--optimize-autoloader] [-a] [--classmap-authoritative] [--apcu-autoloader] [--apcu-autoloader-prefix APCU-AUT
LOADER-PREFIX] [--] [<packages>...]

C:\xampp\php>composer require phpooffice/phpspreadsheet
./composer.json has been created
Running composer update phpooffice/phpspreadsheet
Loading composer repositories with package information
Updating dependencies
Lock file operations: 9 installs, 0 updates, 0 removals
- Locking composer/pcr (3.3.2)
- Locking maennchen/zipstream-php (3.1.2)
- Locking markbaker/complex (3.0.2)
- Locking markbaker/matrix (3.0.1)
- Locking phpooffice/phpspreadsheet (5.2.0)
- Locking psr/http-client (1.0.3)
- Locking psr/http-factory (1.1.0)
- Locking psr/http-message (2.0)
- Locking psr/simple-cache (3.0.0)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 9 installs, 0 updates, 0 removals
- Downloading psr/simple-cache (3.0.0)
- Downloading psr/http-message (2.0)
- Downloading psr/http-factory (1.1.0)
- Downloading psr/http-client (1.0.3)
- Downloading markbaker/matrix (3.0.1)
- Downloading markbaker/complex (3.0.2)
- Downloading maennchen/zipstream-php (3.1.2)
- Downloading composer/pcr (3.3.2)
- Downloading phpooffice/phpspreadsheet (5.2.0)
- Installing psr/simple-cache (3.0.0): Extracting archive
- Installing psr/http-message (2.0): Extracting archive
- Installing psr/http-factory (1.1.0): Extracting archive
- Installing psr/http-client (1.0.3): Extracting archive
- Installing markbaker/matrix (3.0.1): Extracting archive
- Installing markbaker/complex (3.0.2): Extracting archive
- Installing maennchen/zipstream-php (3.1.2): Extracting archive
- Installing composer/pcr (3.3.2): Extracting archive
- Installing phpooffice/phpspreadsheet (5.2.0): Extracting archive
6 package suggestions were added by new dependencies, use `composer suggest` to see details.
Generating autoload files
2 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
No security vulnerability advisories found.
Using version ^5.2 for phpooffice/phpspreadsheet

C:\xampp\php>
```

## Código de prueba:

```
<?php
// =====
// Función: Exporta la lista de empleados desde la BD nomina_db a Excel
// =====

// 1. Importar autoload de Composer (PhpSpreadsheet)
require 'vendor/autoload.php';

// 2. Conectarse a la base de datos
$host = "localhost";
$usuario = "root";
$clave = ""; // sin contraseña
$baseDatos = "nomina_db"; // nombre de la base de datos

$conexion = new mysqli($host, $usuario, $clave, $baseDatos);

if ($conexion->connect_error) {
    die("✗ Error en la conexión: " . $conexion->connect_error);
}

// 3. Importar clases de PhpSpreadsheet
use PhpOffice\PhpSpreadsheet\Spreadsheet;
use PhpOffice\PhpSpreadsheet\Writer\Xlsx;
use PhpOffice\PhpSpreadsheet\Style\Alignment;
use PhpOffice\PhpSpreadsheet\Style\Fill;

// 4. Crear el documento
$documento = new Spreadsheet();
$hoja = $documento->getActiveSheet();
$hoja->setTitle("Empleados");

// 5. Encabezados del reporte
$encabezados = [
    'A1' => 'ID',
    'B1' => 'Nombre',
    'C1' => 'Apellido',
    'D1' => 'Puesto',
    'E1' => 'Salario Base',
    'F1' => 'Sucursal',
    'G1' => 'Fecha de Ingreso',
    'H1' => 'Estatus'
];

foreach ($encabezados as $columna => $texto) {
    $hoja->setCellValue($columna, $texto);
}

// 6. Formato visual para los encabezados
$hoja->getStyle('A1:H1')->applyFromArray([
    'font' => [
        'bold' => true,
        'color' => ['rgb' => 'FFFFFF']
    ],
    'alignment' => [
        'horizontal' => Alignment::HORIZONTAL_CENTER,
        'vertical' => Alignment::VERTICAL_CENTER
    ],
    'fill' => [
        'fillType' => Fill::FILL_SOLID,
        'startColor' => ['rgb' => '4F81BD']
    ]
]);
```



```
// 7. Consultar datos de la tabla empleados
$sql = "
    SELECT
        e.id,
        e.nombre,
        e.apellido,
        p.nombre_puesto AS puesto,
        p.salario_base,
        s.nombre AS sucursal,
        e.fecha_ingreso,
        IF(e.activo = 1, 'Activo', 'Inactivo') AS estatus
    FROM empleados e
    INNER JOIN puestos p ON e.puesto_id = p.id
    INNER JOIN sucursales s ON e.sucursal_id = s.id
    ORDER BY s.nombre, e.apellido;
";

$resultado = $conexion->query($sql);

if (!$resultado) {
    die("❌ Error en la consulta: " . $conexion->error);
}

// 8. Llenar los datos en el Excel
$fila = 2; // inicia debajo de los encabezados

while ($row = $resultado->fetch_assoc()) {
    $hoja->setCellValue('A' . $fila, $row['id']);
    $hoja->setCellValue('B' . $fila, $row['nombre']);
    $hoja->setCellValue('C' . $fila, $row['apellido']);
    $hoja->setCellValue('D' . $fila, $row['puesto']);
    $hoja->setCellValue('E' . $fila, number_format($row['salario_base'], 2));
    $hoja->setCellValue('F' . $fila, $row['sucursal']);
    $hoja->setCellValue('G' . $fila, $row['fecha_ingreso']);
    $hoja->setCellValue('H' . $fila, $row['estatus']);
    $fila++;
}

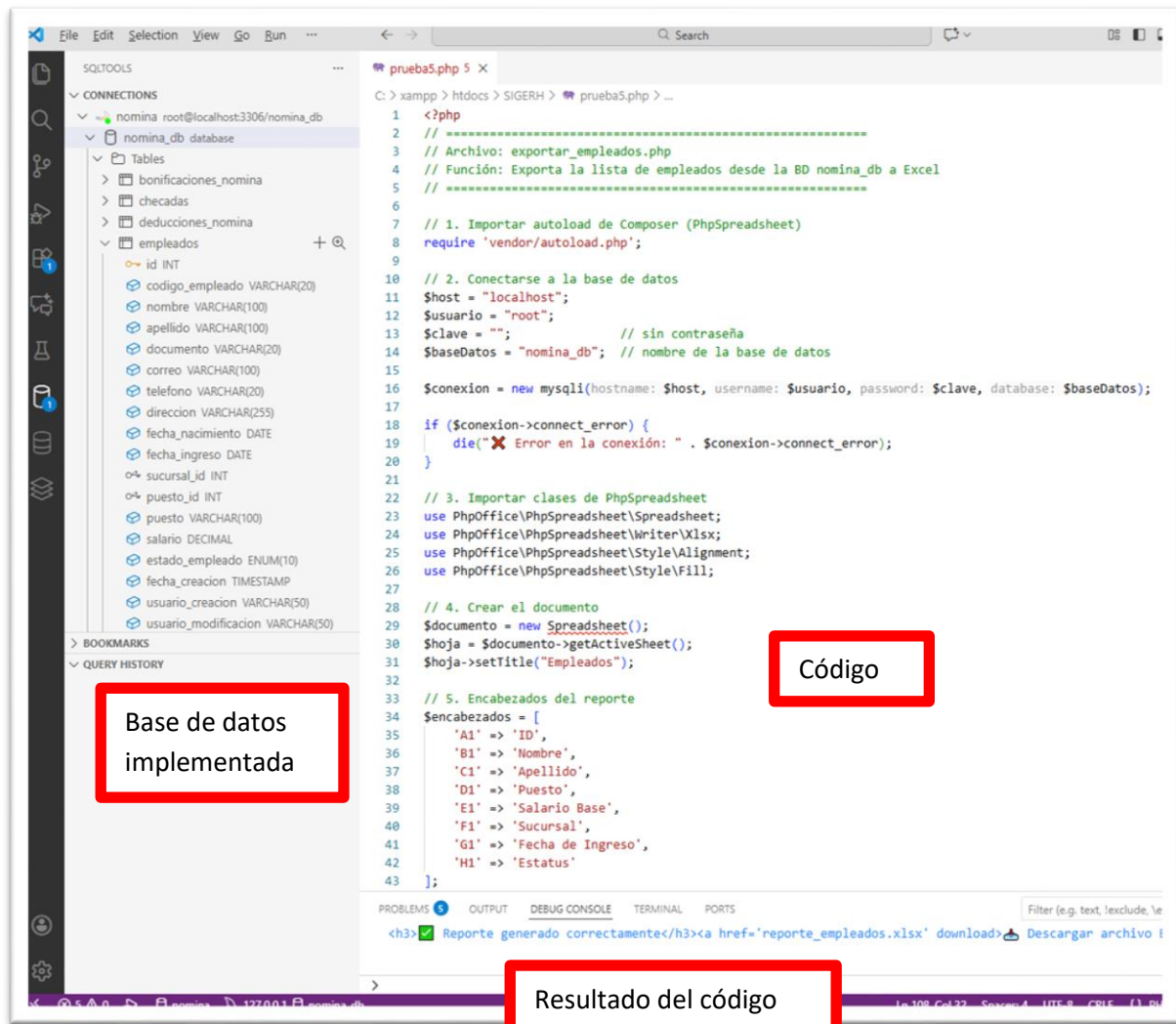
// 9. Ajustar el ancho de las columnas automáticamente
foreach (range('A', 'H') as $col) {
    $hoja->getColumnDimension($col)->setAutoSize(true);
}

// 10. Guardar el archivo Excel
$nombreArchivo = "reporte_empleados.xlsx";
$writer = new Xlsx($documento);
$writer->save($nombreArchivo);

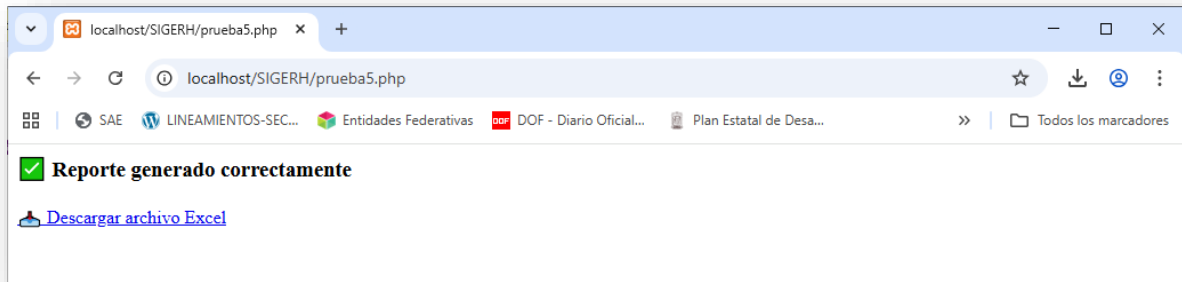
// 11. Mostrar mensaje al usuario
echo "<h3>✔ Reporte generado correctamente</h3>";
echo "<a href='$nombreArchivo' download>📄 Descargar archivo Excel</a>";

// 12. Cerrar la conexión
$conexion->close();
?>
```

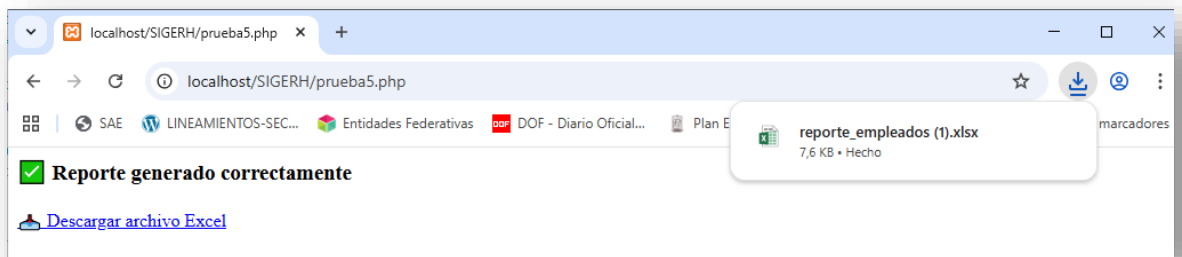
## Pantalla del código:



Desde el navegador:



Al presionar “descargar archivo”, se descarga un archivo es Excel que corresponde a los empleados registrados en la base de datos, como se muestra a continuación:



El archivo que se genera es el siguiente:

A screenshot of an Excel spreadsheet titled 'reporte\_empleados (1) - Excel'. The spreadsheet contains a table with 8 columns: ID, Nombre, Apellido, Puesto, Salario Base, Sucursal, Fecha de Ingreso, and Estatus. The data is organized into 26 rows, with the first row being the header and the subsequent rows containing employee records.

ID	Nombre	Apellido	Puesto	Salario Base	Sucursal	Fecha de Ingreso	Estatus
1	Juan	Hernández	Analista de Nómina	15,000.00	Sucursal Central	2022-03-01	Activo
2	19	Ricardo	Lara	20,000.00	Sucursal Central	2020-11-09	Activo
3	4	Lucía	Martínez	15,000.00	Sucursal Central	2023-05-10	Activo
4	13	Andrés	Pérez	15,000.00	Sucursal Central	2022-08-15	Activo
5	22	Leticia	Ramos	18,000.00	Sucursal Central	2019-04-13	Activo
6	7	Jorge	Ramírez	20,000.00	Sucursal Central	2020-09-10	Activo
7	16	Adriana	Reyes	15,000.00	Sucursal Central	2023-02-20	Activo
8	10	Claudia	Ríos	15,000.00	Sucursal Central	2022-06-11	Activo
9	20	Estefanía	Campos	15,000.00	Sucursal Norte	2023-07-11	Activo
10	8	Patricia	Cruz	15,000.00	Sucursal Norte	2019-03-01	Activo
11	5	Pedro	Gómez	18,000.00	Sucursal Norte	2021-07-01	Activo
12	17	David	Mendoza	18,000.00	Sucursal Norte	2022-01-25	Activo
13	11	Luis	Morales	20,000.00	Sucursal Norte	2018-11-12	Activo
14	14	Rodrigo	Navarro	12,000.00	Sucursal Norte	2021-09-19	Activo
15	23	Hector	Silva	20,000.00	Sucursal Norte	2020-03-30	Activo
16	12	Carmen	Flores	18,000.00	Sucursal Sur	2023-04-01	Activo
17	21	Tomás	Herrera	12,000.00	Sucursal Sur	2022-09-09	Activo
18	3	Carlos	Pérez	20,000.00	Sucursal Sur	2024-06-10	Activo
19	9	Fernando	Santos	12,000.00	Sucursal Sur	2021-04-23	Activo
20	18	Gabriela	Suárez	15,000.00	Sucursal Sur	2021-05-05	Activo
21	6	Ana	Torres	12,000.00	Sucursal Sur	2022-02-18	Activo
22	15	Sergio	Vega	20,000.00	Sucursal Sur	2020-10-25	Activo
23							
24							
25							
26							





## 10.2 PHPMailer (Envío de Correos Electrónicos)

### Descripción

PHPMailer es una de las librerías más utilizadas en PHP para enviar correos electrónicos con autenticación SMTP, adjuntos, plantillas HTML y notificaciones automatizadas. Su robustez y facilidad de integración la hacen ideal para sistemas administrativos que requieren comunicación interna y externa.

### Justificación

En el sistema SIGERH, el envío de correos es un componente clave para automatizar procesos como:

-  Notificaciones de aprobación/rechazo de permisos o vacaciones.
-  Envío de avisos de nómina o recibos PDF.
-  Recuperación de contraseña para usuarios del sistema.
-  Alertas al área de Recursos Humanos sobre incidencias capturadas.

PHPMailer permite:

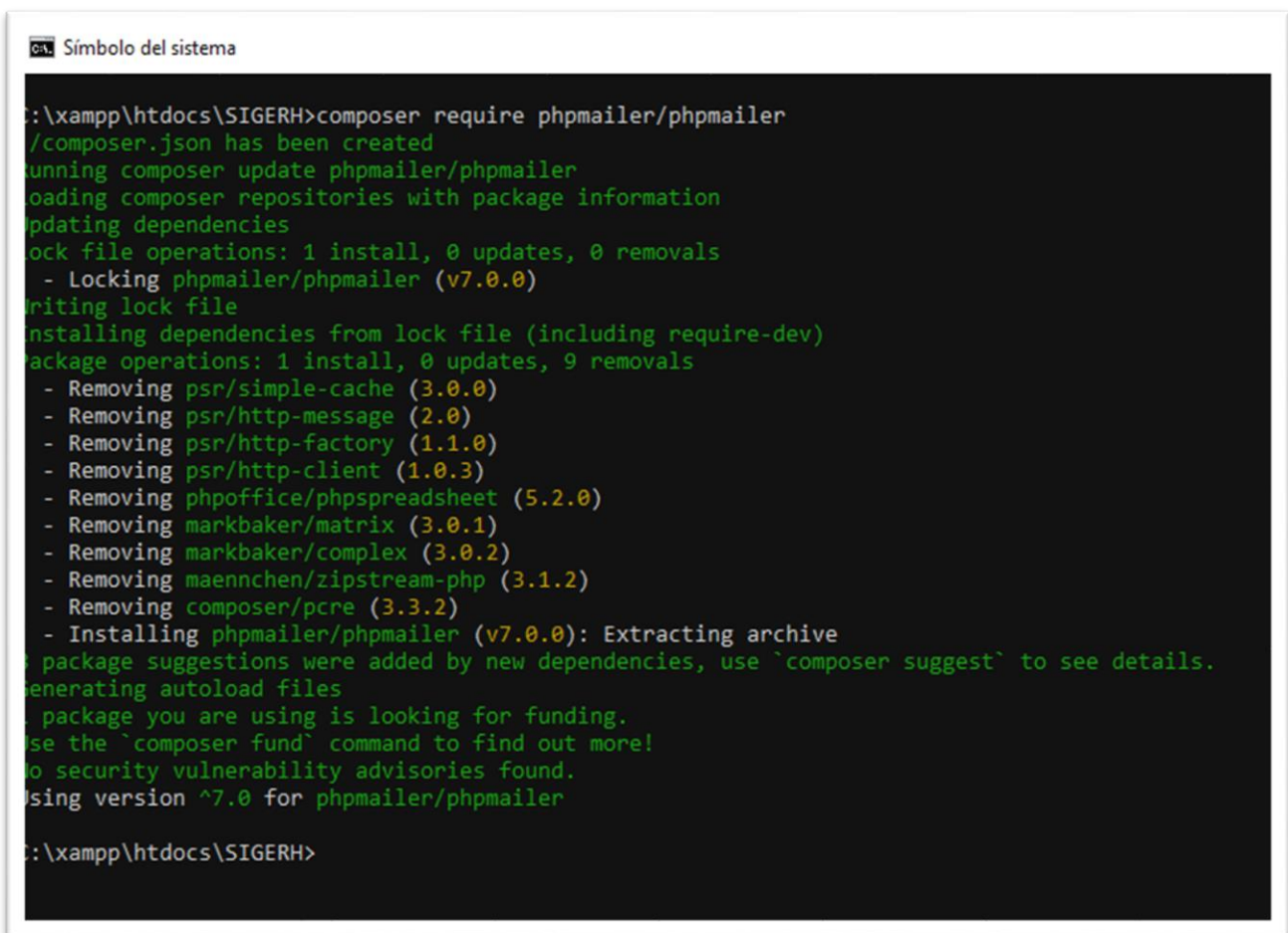
- ✓ Enviar correos de forma segura mediante SMTP.
- ✓ Adjuntar documentos como recibos de nómina o justificaciones digitalizadas.
- ✓ Usar plantillas HTML para mensajes más profesionales.

## Instalación de la librería:

PhpMailer se instaló mediante el uso de composer, usando el comando:

**composer require phpmailer/phpmailer**

En la siguiente pantalla se muestra el proceso de instalación:



```
C:\> Símbolo del sistema

C:\xampp\htdocs\SIGERH>composer require phpmailer/phpmailer
./composer.json has been created
Running composer update phpmailer/phpmailer
Loading composer repositories with package information
Updating dependencies
Lock file operations: 1 install, 0 updates, 0 removals
  - Locking phpmailer/phpmailer (v7.0.0)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 1 install, 0 updates, 9 removals
  - Removing psr/simple-cache (3.0.0)
  - Removing psr/http-message (2.0)
  - Removing psr/http-factory (1.1.0)
  - Removing psr/http-client (1.0.3)
  - Removing phpooffice/phpspreadsheet (5.2.0)
  - Removing markbaker/matrix (3.0.1)
  - Removing markbaker/complex (3.0.2)
  - Removing maennchen/zipstream-php (3.1.2)
  - Removing composer/pcre (3.3.2)
  - Installing phpmailer/phpmailer (v7.0.0): Extracting archive
Some package suggestions were added by new dependencies, use `composer suggest` to see details.
Generating autoload files
The package you are using is looking for funding.
Use the `composer fund` command to find out more!
No security vulnerability advisories found.
Using version ^7.0 for phpmailer/phpmailer

C:\xampp\htdocs\SIGERH>
```

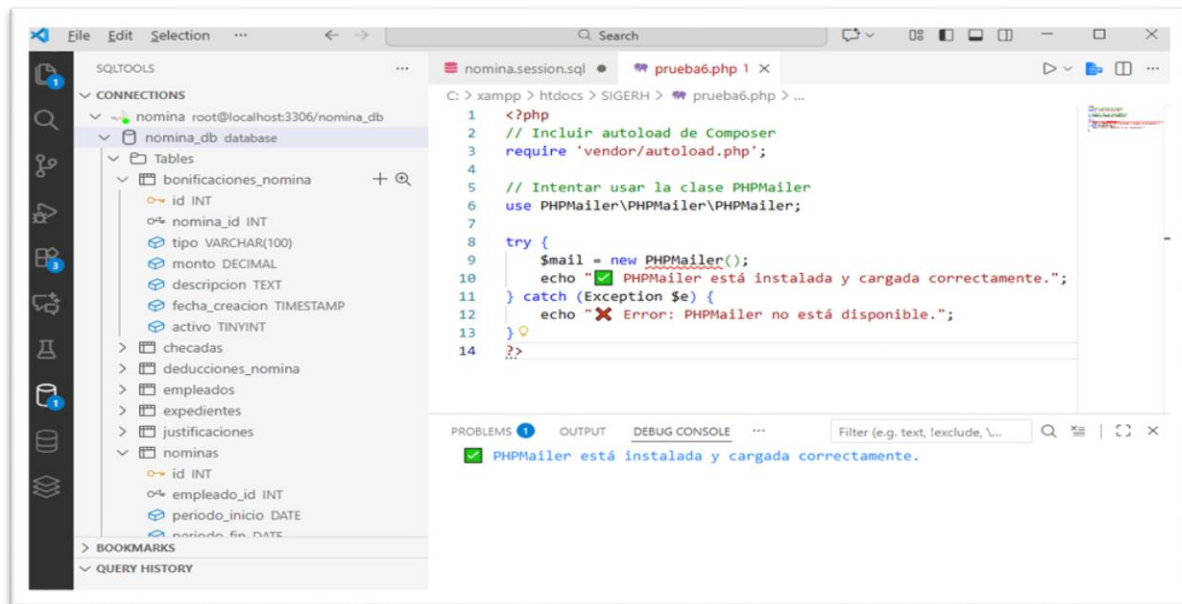
## Código de prueba:

```
<?php
// Incluir autoload de Composer
require 'vendor/autoload.php';

// Intentar usar la clase PHPMailer
use PHPMailer\PHPMailer\PHPMailer;

try {
    $mail = new PHPMailer();
    echo "✓ PHPMailer está instalada y cargada correctamente.";
} catch (Exception $e) {
    echo "✗ Error: PHPMailer no está disponible.";
}
?>
```

## Pantalla del código



## Desde el navegador:






### 10.3 TCPDF (Generación de documentos PDF con cabecera, pie y logo)

#### Descripción

TCPDF es una librería PHP que permite generar documentos PDF directamente desde PHP, incluyendo soporte para imágenes, encabezados, pies de página, tablas, estilos de texto y contenido dinámico. Es muy útil para crear recibos de nómina, reportes de empleados y otros documentos administrativos de manera profesional.

#### Justificación

En SIGERH, se requiere generar documentos formales y profesionalmente presentados, como:

-  Recibos de nómina en PDF
-  Reportes de empleados
-  Documentación contable

TCPDF permite personalizar el documento con:

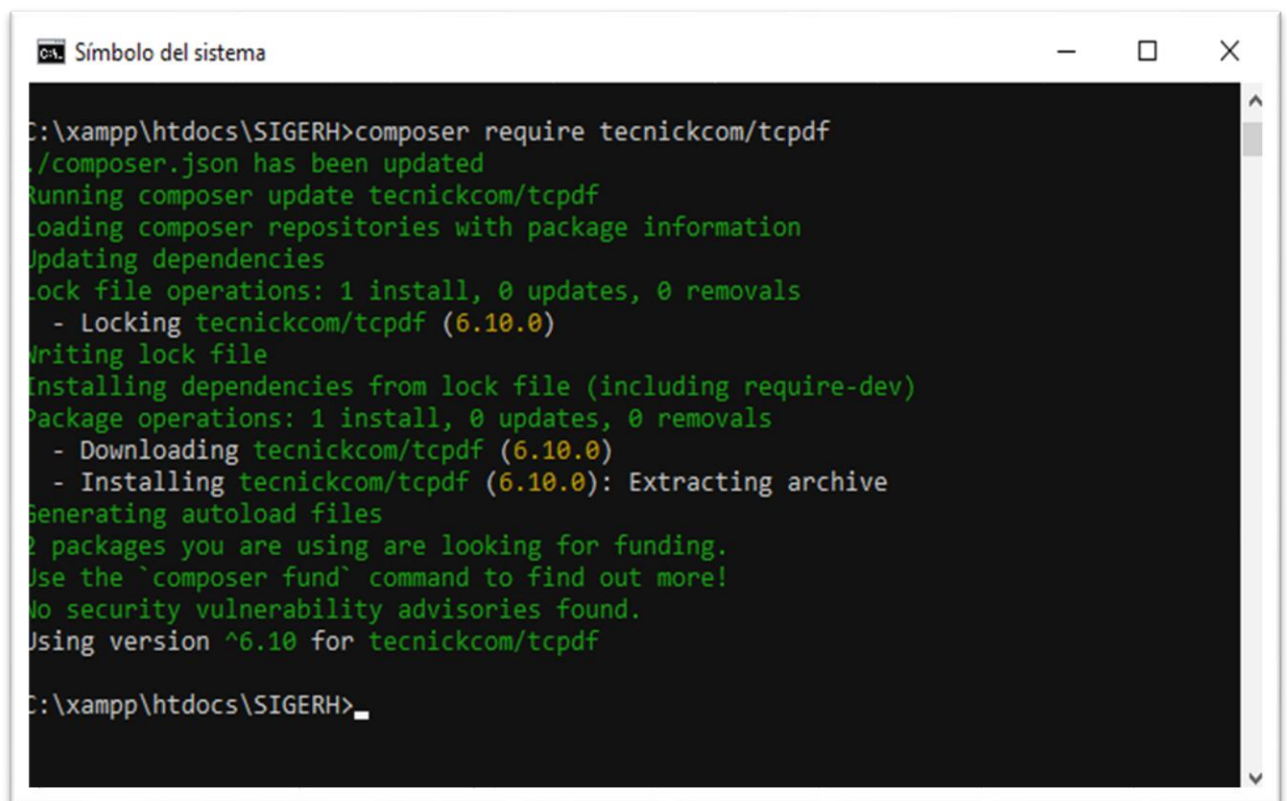
- ✓ logo de la empresa
- ✓ encabezados
- ✓ pies de página
- ✓ estilos de celda
- ✓ tablas
- ✓ contenido dinámico desde la base de datos.

## Instalación de la librería:

PhpMailer se instaló mediante el uso de composer, usando el comando:

**composer require tecnickcom/tcpdf**

En la siguiente pantalla se muestra el proceso de instalación:



```
C:\xampp\htdocs\SIGERH>composer require tecnickcom/tcpdf
./composer.json has been updated
Running composer update tecnickcom/tcpdf
Loading composer repositories with package information
Updating dependencies
Lock file operations: 1 install, 0 updates, 0 removals
  - Locking tecnickcom/tcpdf (6.10.0)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
  - Downloading tecnickcom/tcpdf (6.10.0)
  - Installing tecnickcom/tcpdf (6.10.0): Extracting archive
Generating autoload files
2 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
No security vulnerability advisories found.
Using version ^6.10 for tecnickcom/tcpdf

C:\xampp\htdocs\SIGERH>
```



## Código de prueba:

```
<?php
// =====
// Función: Genera un PDF con el resumen de nómina por empleado
// =====

// 1. Importar TCPDF
require_once('vendor/autoload.php');

// 2. Conectarse a MySQL
$host = "localhost";
$usuario = "root";
$clave = "";
$baseDatos = "nomina_db";

$conexion = new mysqli($host, $usuario, $clave, $baseDatos);

if ($conexion->connect_error) {
    die("✗ Error en la conexión: " . $conexion->connect_error);
}

// 3. Consulta
$sql = "
SELECT
    s.nombre AS sucursal,
    e.apellido AS apellido,
    e.nombre AS nombre,
    p.nombre_puesto AS puesto,
    p.salario_base AS salario,
    IFNULL(SUM(b.monto), 0) AS total_bonificaciones,
    IFNULL(SUM(d.monto), 0) AS total_deducciones,
    (p.salario_base + IFNULL(SUM(b.monto), 0) - IFNULL(SUM(d.monto), 0)) AS salario_neto
FROM empleados e
INNER JOIN puestos p
    ON e.puesto_id = p.id
INNER JOIN sucursales s
    ON e.sucursal_id = s.id
LEFT JOIN nominas n
    ON n.empleado_id = e.id
LEFT JOIN bonificaciones_nomina b
    ON b.nomina_id = n.id
LEFT JOIN deducciones_nomina d
    ON d.nomina_id = n.id
WHERE e.activo = 1
```

```

GROUP BY s.nombre, e.apellido, e.nombre, p.nombre_puesto, p.salario_base
ORDER BY s.nombre ASC, e.apellido ASC;
";

$resultado = $conexion->query($sql);

if (!$resultado) {
    die("✗ Error en la consulta SQL: " . $conexion->error);
}

// 4. Crear PDF
$pdf = new TCPDF();
$pdf->SetCreator(PDF_CREATOR);
$pdf->SetAuthor('SIGERH');
$pdf->SetTitle('Resumen de Nómina');

$pdf->SetHeaderData('logo.png', 30, 'SIGERH - Recursos Humanos', 'Resumen de Nómina');
$pdf->setHeaderFont(['helvetica', '', 11]);
$pdf->setFooterFont(['helvetica', '', 9]);
$pdf->SetMargins(12, 35, 12);
$pdf->SetAutoPageBreak(TRUE, 20);

$pdf->AddPage();
$pdf->SetFont('helvetica', '', 10);

// 5. Construcción HTML
$html = '
<h2 style="text-align:center;">Resumen de Nómina</h2>
<table border="1" cellpadding="4">
<tr style="background-color:#4F81BD;color:white;font-weight:bold;text-align:center;">
    <th>Sucursal</th>
    <th>Apellido</th>
    <th>Nombre</th>
    <th>Puesto</th>
    <th>Salario Base</th>
    <th>Bonificaciones</th>
    <th>Deducciones</th>
    <th>Salario Neto</th>
</tr>
';

while ($row = $resultado->fetch_assoc()) {

    $html .= '
    <tr>

```

```
<td>'.htmlspecialchars($row['sucursal']).'</td>
<td>'.htmlspecialchars($row['apellido']).'</td>
<td>'.htmlspecialchars($row['nombre']).'</td>
<td>'.htmlspecialchars($row['puesto']).'</td>
<td>$.number_format($row['salario'], 2).'</td>
<td>$.number_format($row['total_bonificaciones'], 2).'</td>
<td>$.number_format($row['total_deducciones'], 2).'</td>
<td><b>$.number_format($row['salario_neto'], 2).'</b></td>
</tr>;
}

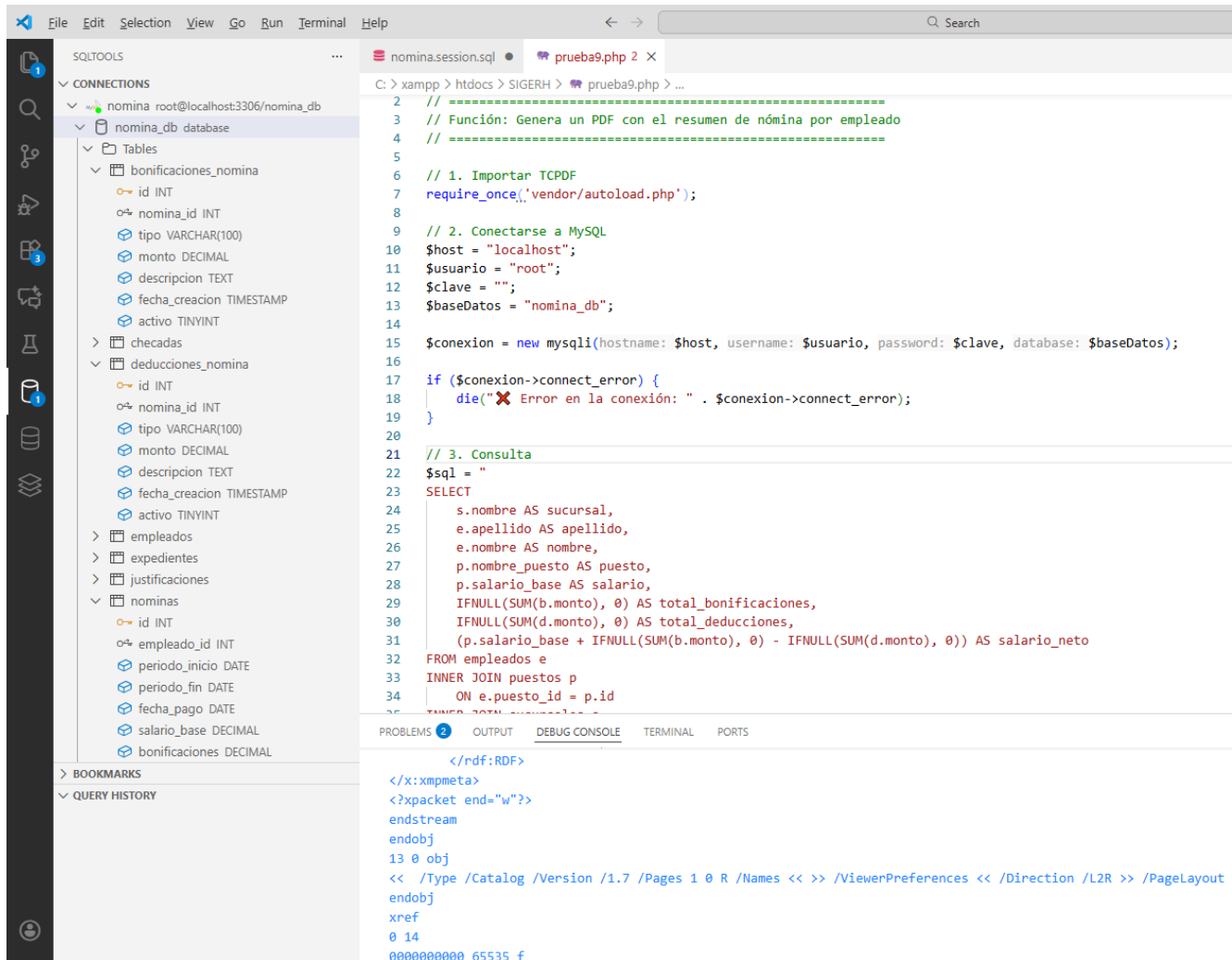
$html .= '</table>';

// 6. Pasar HTML al PDF
$pdf->writeHTML($html, true, false, true, false, '');

// 7. Mostrar PDF
$pdf->Output('reporte_nomina_resumen.pdf', 'I');

// 8. Cerrar conexión
$conexión->close();
?>
```

## Pantalla del código



Desde el navegador:

Resumen de Nómina

localhost/sigerh/prueba9.php

SAE LINEAMIENTOS-SEC... Entidades Federativas DOF - Diario Oficial... Plan Estatal de Desa... Todos los marcadores

Resumen de Nómina 1 / 2 110%

**SIGERH - Recursos Humanos**  
Resumen de Nómina

**Resumen de Nómina**

Sucursal	Apellido	Nombre	Puesto	Salario Base	Bonificaciones	Deducciones	Salario Neto
Sucursal Central	Hernandez	Juan	Analista de Nmina	\$15,000.00	\$1,000.00	\$1,200.00	<b>\$14,800.00</b>
Sucursal Central	Lara	Ricardo	Contador General	\$20,000.00	\$1,500.00	\$1,150.00	<b>\$20,350.00</b>
Sucursal Central	Martnez	Luca	Analista de Nmina	\$15,000.00	\$800.00	\$1,000.00	<b>\$14,800.00</b>
Sucursal Central	Prez	Andrs	Analista de Nmina	\$15,000.00	\$800.00	\$1,200.00	<b>\$14,600.00</b>
Sucursal Central	Ramos	Leticia	Encargado de Sucursal	\$18,000.00	\$0.00	\$0.00	<b>\$18,000.00</b>
Sucursal Central	Ramrez	Jorge	Contador General	\$20,000.00	\$1,500.00	\$1,100.00	<b>\$20,400.00</b>
Sucursal Central	Reyes	Adriana	Analista de Nmina	\$15,000.00	\$1,000.00	\$1,000.00	<b>\$15,000.00</b>
Sucursal Central	Ros	Claudia	Analista de Nmina	\$15,000.00	\$900.00	\$1,300.00	<b>\$14,600.00</b>
Sucursal Norte	Campos	Estefana	Analista de Nmina	\$15,000.00	\$900.00	\$500.00	<b>\$15,400.00</b>
Sucursal Norte	Cruz	Patricia	Analista de Nmina	\$15,000.00	\$400.00	\$400.00	<b>\$15,000.00</b>
Sucursal Norte	Gmez	Pedro	Encargado de Sucursal	\$18,000.00	\$500.00	\$600.00	<b>\$17,900.00</b>
Sucursal Norte	Mendoza	David	Encargado de Sucursal	\$18,000.00	\$1,200.00	\$600.00	<b>\$18,600.00</b>
Sucursal Norte	Morales	Luis	Contador General	\$20,000.00	\$1,000.00	\$550.00	<b>\$20,450.00</b>

## 11. CONCLUSIONES

La implementación del Sistema Integral de Gestión de Recursos Humanos (SIGERH) representa un avance significativo en la modernización de los procesos administrativos y operativos. A lo largo del desarrollo se logró diseñar, estructurar y poner en funcionamiento una solución tecnológica que responde directamente a las necesidades diagnosticadas al inicio: control ineficiente de información, procesos manuales, duplicidad de datos, errores frecuentes en cálculos de nómina y escasa trazabilidad en los movimientos relacionados con los empleados.

El análisis y construcción de la base de datos permitió establecer un modelo sólido que garantiza integridad, consistencia y escalabilidad de la información. Asimismo, la integración de librerías profesionales, como TCPDF, PhpSpreadsheet y PHPMailer, fortaleció las capacidades del sistema para generar reportes, exportar información y producir documentación formal, facilitando la comunicación interna y mejorando la calidad en la presentación de datos.

Durante el desarrollo se demostró que la automatización no solo reduce tiempos y errores, sino que abre la puerta a una mejor toma de decisiones basada en información confiable y actualizada. El SIGERH permite consolidar en un solo lugar datos esenciales que antes estaban dispersos o gestionados de manera informal, lo que se traduce en mayor transparencia, mejor control administrativo y mayor capacidad de respuesta ante auditorías, revisiones o necesidades operativas.

En conjunto, el desarrollo del SIGERH demuestra la importancia de aplicar metodologías de ingeniería de software, análisis sistemático de procesos y herramientas tecnológicas modernas en la gestión del talento humano. La solución entregada cumple con los objetivos planteados, mejora sustancialmente la eficiencia operativa y establece un estándar más alto de profesionalización en la administración de recursos humanos. Este proyecto no solo responde a una necesidad actual, sino que contribuye al fortalecimiento institucional y al desarrollo de una cultura organizacional basada en la precisión, la transparencia y la mejora continua.

## 12. BIBLIOGRAFÍA

MySQL. (2024). MySQL 8.0 Reference Manual. Oracle Corporation.  
<https://dev.mysql.com/doc/>

Pressman, R. S., & Maxim, B. R. (2020). Software engineering: A practitioner's approach (9.<sup>a</sup> ed.). McGraw-Hill.

Rumbaugh, James; Jacobson, Ivar; Booch, Grady. El proceso unificado de desarrollo de software. Addison-Wesley. Madrid, España, 2000

Pons, Claudia; Giandini, Roxana; Pérez, Gabriela. Desarrollo de Software Dirigido por Modelos. Teorías, Metodologías y Herramientas. McGraw-Hill Education, 2010, p. 978-950.